IN THE CLAIMS

The following is a complete listing of the claims. This listing replaces all earlier versions and listings of the claims.

Claim 1 (currently amended): A method of generating instructions for a directed ~~adjacency~~ acyclic graph, [[said]] the directed ~~adjacency~~ acyclic graph comprising one or more parent nodes and one or more leaf nodes, each ~~of which said~~ parent node representing an operator and having branches to respective descendent nodes, and each ~~of which said~~ leaf node representing a graphic object having object edges, said method comprising the steps of:

determining groups of one or more pixel locations, wherein the groups are bounded by the object edges;

determining, for each [[said]] group, a portion of [[said]] the directed ~~adjacency~~ acyclic graph in accordance with activities of the operators, wherein the [[said]] portion of the directed ~~adjacency~~ acyclic graph is that portion which passes data up the directed ~~adjacency~~ acyclic graph; and

generating, for each [[said]] group, instructions for the determined portion of the directed ~~adjacency~~ acyclic graph, wherein operator instructions are generated for those operators of the determined portion of the directed ~~adjacency~~ acyclic graph having active branches and wherein leaf instructions are generated for those graphic objects which are active at [[said]] the group of one or more pixel locations.

Claim 2 (currently amended): [[The]] A method ~~as claimed in~~ according to claim 1, wherein said method further comprises the step of generating a table for storing data concerning the activity of the operators and the activity of the branches of the parent nodes.

Claim 3 (currently amended): [[The]] A method ~~as claimed in~~ according to claim 2, wherein said determining step ~~comprises~~ includes the sub-step of updating, for each [[said]] group, the [[said]] data stored in [[said]] the generated table.

Claim 4 (currently amended): [[The]] A method ~~as claimed in~~ according to claim 1, wherein said step of generating instructions ~~comprises a~~ includes the sub-step of traversing [[said]] the determined portion of the directed ~~adjacency~~ acyclic graph and generating instructions for the active operators and the leaf value instructions in [[said]] the determined portion of the directed ~~adjacency~~ acyclic graph.

Claim 5 (currently amended): [[The]] A method ~~as claimed in~~ according to claim 1, wherein [[said]] the directed ~~adjacency~~ acyclic graph is an expression tree.

Claim 6 (currently amended): [[The]] A method ~~as claimed in~~ according to claim 1, wherein [[said]] the parent nodes represent binary operators.

Claim 7 (currently amended): A method of generating instructions for an expression tree, [[said]] the expression tree having a plurality of nodes comprising one or

more binary nodes and a plurality of leaf nodes, wherein each [[said]] binary node ~~having~~ has a left branch to a descendent [[said]] node and a right branch to [[a]] another descendent [[said]] node and ~~representing~~ represents a binary operation on [[said]] the two descendant nodes, and wherein each [[said]] node represents a graphic object having object edges, with one or more [[said]] of the graphic objects overlapping, ~~each said~~ the overlapping graphics objects comprising a left node region, a common region, and a right node region, said method comprising the steps of:

determining groups of one or more pixel locations, wherein the groups are bounded by the object edges;

determining, for each [[said]] group, whether the left and right branches of [[said]] the one or more binary nodes are active or inactive;

traversing, for each [[said]] group, [[said]] the expression tree, wherein the left branch of any previously traversed [[said]] binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] left node region is required for the binary operation of [[said]] the previously traversed binary node and the left branch is active and the right branch is inactive of [[said]] the previously traversed binary node, and wherein a right branch of any previously traversed binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] right node region is required for the binary operation of [[said]] the previously traversed binary node and the right branch is active and the left branch is inactive of [[said]] the previously traversed binary node; and

generating, for each [[said]] group, operator instructions for any [[said]] traversed binary node having active [[said]] right and left branches, and leaf value instructions for any traversed leaf node.

Claim 8 (currently amended): A method ~~as claimed in~~ according to claim 7, wherein said ~~tranversing~~ traversing step traverses [[said]] the expression tree, wherein the left branch of any previously traversed [[said] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node, and wherein the right branch of any previously traversed [[said]] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node.

Claim 9 (currently amended): A method of rendering an expression tree into a raster pixel image having a plurality of scanlines and a plurality of pixel locations on each

[[said]] scanline, [[said]] the expression tree having a plurality of nodes comprising one or more binary nodes and a plurality of leaf nodes, wherein each [[said]] binary node ~~having~~ has a left branch to a descendent [[said]] node and a right branch to [[a]] another descendent [[said]] node and ~~representing~~ represents a binary operation on [[said]] the two descendant nodes, and wherein each [[said]] node represents a graphic object having object edges, with one or more [[said]] graphic objects overlapping, ~~each said~~ the overlapping graphics objects comprising a left node region, a common region, and a right node region, said method comprising the steps of:

generating a table representative of [[said]] the expression tree, wherein [[said]] the table comprises a plurality of records corresponding to associated [[said]] binary nodes and leaf nodes, and each [[said]] record of ~~a said~~ an associated binary node comprises a first field indicating whether a [[said]] left region is required for operation of the operator of [[said]] the associated binary node, a second field indicating whether a right region is required for operation of the operator of [[said]] the associated binary node, a third field capable of indicating whether a [[said]] left branch of [[said]] the associated binary node is active, and a fourth field capable of indicating whether a [[said]] right branch of [[said]] the associated binary node is active;

determining groups of one or more pixel locations, wherein the groups are bounded by object edges;

determining, for each [[said]] group, whether the left and right branches of [[said]] the one or more binary nodes are active or inactive;

updating, for each [[said]] group, [[said]] the third and fourth fields of [[said]] the table for [[said]] the determined active and inactive branches;

traversing, for each [[said]] group, [[said]] the expression tree in accordance with [[said]] the updated table wherein the left branch of any previously traversed [[said]] binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] left node region is required for the binary operation of said previously traversed binary node and the left branch is active and the right branch is inactive of [[said]] the previously traversed binary node, and wherein a right branch of any previously traversed binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] right node region is required for the binary operation of [[said]] the previously traversed binary node and the right branch is active and the left branch is inactive of [[said]] the previously traversed binary node;

generating, for each [[said]] group, operator instructions for any [[said]] traversed binary node having active said right and left branches, and leaf value instructions for any traversed leaf node; and

executing, for each [[said]] group, corresponding [[said]] generated instructions so as to render [[said]] the image.

Claim 10 (currently amended): A method ~~as claimed in~~ according to claim 9,

wherein [[said]] the table further comprises for each [[said]] record of ~~a said~~ an associated binary node a fifth field indicating whether the graphic object representing [[said]] the descendent node on the right branch of [[said]] the associated binary node obscures the graphic object representing [[said]] the descendent node on the left branch of

[[said]] the associated binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the associated binary node, and a sixth field indicating whether the graphic object representing [[said]] the descendent node on the left branch of [[said]] the associated binary node obscures the graphic object representing [[said]] the descendent node on the right branch of [[said]] the associated binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the associated binary node[[;]], and

wherein said ~~tranversing~~ traversing step traverses [[said]] the expression tree in accordance with [[said]] the updated table, wherein the left branch of any previously traversed [[said]] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node, and wherein the right branch of any previously traversed [[said]] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node.

Claim 11 (currently amended): A method ~~as claimed in~~ according to claim 10, wherein [[said]] the fifth and sixth fields are used to implement a CLIP IN or a CLIP OUT operation.

Claim 12 (currently amended): ~~Apparatus~~ An apparatus for generating instructions for a directed ~~adjacency~~ acyclic graph, [[said]] the directed ~~adjacency~~ acyclic graph comprising one or more parent nodes and one or more leaf nodes, each ~~of which said~~ parent node representing an operator and having branches to respective descendent nodes, and each ~~of which said~~ leaf node representing a graphic object having object edges, said apparatus comprising:

means for determining groups of one or more pixel locations, wherein the groups are bounded by the object edges;

means for determining, for each [[said]] group, a portion of [[said]] the directed ~~adjacency~~ acyclic graph in accordance with activities of the operators, wherein the [[said]] portion of the directed ~~adjacency~~ acyclic graph is that portion which passes data up the directed ~~adjacency~~ acyclic graph; and

means for generating, for each [[said]] group, instructions for the determined portion of the directed ~~adjacency~~ acyclic graph, wherein operator instructions are generated for those operators of the determined portion of the directed ~~adjacency~~ acyclic graph having active branches and wherein leaf instructions are generated for those graphic objects which are active at [[said]] the group of one or more pixel locations.

Claim 13 (currently amended): ~~Apparatus as claimed in~~ An apparatus according to claim 12, wherein said apparatus further comprises means for generating a table for storing data concerning the activity of the operators and the activity of the branches of the parent nodes.

Claim 14 (currently amended): ~~Apparatus as claimed in~~ An apparatus according to claim 13, wherein said means for determining [[means]] groups comprises means for updating, for each [[said]] group, the [[said]] data stored in [[said]] the generated table.

Claim 15 (currently amended): ~~Apparatus as claimed in~~ An apparatus according to claim 12, wherein said means for generating instructions comprises means for traversing [[said]] the portion of the directed ~~adjacency~~ acyclic graph and means for generating instructions for active operators and leaf value instructions in [[said]] the portion of the directed ~~adjacency~~ acyclic graph.

Claim 16 (currently amended): ~~Apparatus as claimed in~~ An apparatus according to claim 12, wherein [[said]] the directed ~~adjacency~~ acyclic graph is an expression tree.

Claim 17 (currently amended): ~~Apparatus as claimed in~~ An apparatus according to claim 12, wherein [[said]] the parent nodes represent binary operators.

Claim 18 (currently amended): ~~Apparatus~~ An apparatus for generating instructions for an expression tree, [[said]] the expression tree having a plurality of nodes comprising one or more binary nodes and a plurality of leaf nodes, wherein each [[said]] binary node ~~having~~ has a left branch to a descendent [[said]] node and a right branch to [[a]] another descendent [[said]] node and ~~representing~~ represents a binary operation on [[said]] the two descendant nodes, and wherein each [[said]] node represents a graphic object having object edges, with one or more [[said]] graphic objects overlapping, ~~each said~~ the overlapping graphics objects comprising a left node region, a common region, and a right node region, said apparatus comprising:

means for determining groups of one or more pixel locations, wherein the groups are bounded by the object edges;

means for determining, for each [[said]] group, whether the left and right branches of [[said]] one or more binary nodes are active or inactive;

means for traversing, for each [[said]] group, [[said]] the expression tree, wherein the left branch of any previously traversed [[said]] binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] left node region is required for the binary operation of [[said]] the previously traversed binary node and the left branch is active and the right branch is inactive of [[said]] the previously traversed binary node, and wherein a right branch of any previously traversed binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] right node region is required for the

binary operation of [[said]] the previously traversed binary node and the right branch is active and the left branch is inactive of [[said]] the previously traversed binary node; and

means for generating, for each [[said]] group, operator instructions for any [[said]] traversed binary node having active [[said]] right and left branches, and leaf value instructions for any traversed leaf node.

Claim 19 (currently amended): ~~Apparatus as claimed in~~ An apparatus according to claim 18, wherein said ~~tranversing~~ traversing means traverses [[said]] the expression tree, wherein the left branch of any previously traversed said binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node, and wherein the right branch of any previously traversed [[said]] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node.

Claim 20 (currently amended): ~~Apparatus~~ An apparatus for rendering an expression tree into a raster pixel image having a plurality of scanlines and a plurality of pixel locations on each [[said]] scanline, [[said]] the expression tree having a plurality of nodes comprising one or more binary nodes and a plurality of leaf nodes, wherein each [[said]] binary node ~~having~~ has a left branch to a descendent [[said]] node and a right branch to [[a]] another descendent [[said]] node and ~~representing~~ represents a binary operation on [[said]] the two descendant nodes, and wherein each [[said]] node represents a graphic object having object edges, with one or more [[said]] graphic objects overlapping, ~~each said~~ the overlapping graphics objects comprising a left node region, a common region, and a right node region, said apparatus comprising:

means for generating a table representative of [[said]] the expression tree, wherein [[said]] the table comprises a plurality of records corresponding to associated [[said]] binary nodes and leaf nodes, and each [[said]] record of ~~a said~~ an associated binary node comprises a first field indicating whether a [[said]] left region is required for operation of the operator of [[said]] the associated binary node, a second field indicating whether a right region is required for operation of the operator of [[said]] the associated binary node, a third field capable of indicating whether a [[said]] left branch of [[said]] the associated binary node is active, and a fourth field capable of indicating whether a [[said]] right branch of [[said]] the associated binary node is active;

means for determining groups of one or more pixel locations, wherein te groups are bounded by the object edges;

means for determining, for each [[said]] group, whether the left and right branches of [[said]] the one or more binary nodes are active or inactive;

means for updating, for each [[said]] group, [[said]] the third and fourth fields of [[said]] the table for [[said]] the determined active and inactive branches;

means for traversing, for each [[said]] group, [[said]] the expression tree in accordance with [[said]] the updated table wherein the left branch of any previously traversed [[said]] binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] left node region is required for the binary operation of [[said]] the previously traversed binary node and the left branch is active and the right branch is inactive of [[said]] the previously traversed binary node, and wherein a right branch of any previously traversed binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] right node region is required for the binary operation of [[said]] the previously traversed binary node and the right branch is active and the left branch is inactive of [[said]] the previously traversed binary node;

means for generating, for each [[said]] group, operator instructions for any [[said]] traversed binary node having active said right and left branches, and leaf value instructions for any traversed leaf node; and

means for executing, for each [[said]] group, corresponding [[said]] generated instructions so as to render [[said]] the image.

Claim 21 (currently amended): ~~Apparatus as claimed in~~ An apparatus according to claim 20,

wherein [[said]] the table further comprises for each [[said]] record of ~~a said~~ an associated binary node a fifth field indicating whether the graphic object representing [[said]] the descendent node on the right branch of [[said]] the associated binary node obscures the graphic object representing [[said]] the descendent node on the left branch of [[said]] the associated binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the associated binary node, and a sixth field indicating whether the graphic object representing [[said]] the descendent node on the left branch of [[said]] the associated binary node obscures the graphic object representing [[said]] the descendent node on the right branch of [[said]] the associated binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the associated binary node[[;]], and

wherein said ~~tranversing~~ traversing means traverses [[said]] the expression tree in accordance with [[said]] the updated table, wherein the left branch of any previously traversed [[said]] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node, and wherein the right branch of any previously traversed [[said]] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the left branch of [[said]] the

previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node.

Claim 22 (currently amended): ~~Apparatus as claimed in~~ An apparatus according to claim 21, wherein [[said]] the fifth and sixth fields are used to implement a CLIP IN or a CLIP OUT operation.

Claim 23 (currently amended): A computer readable medium comprising a computer program for generating instructions for a directed ~~adjacency~~ acyclic graph, [[said]] the directed ~~adjacency~~ acyclic graph comprising one or more parent nodes and one or more leaf nodes, each ~~of which said~~ parent node representing an operator and having branches to respective descendent nodes, and each ~~of which said~~ leaf node representing a graphic object having object edges, said computer program comprising:

code for determining groups of one or more pixel locations, wherein the groups are bounded by the object edges;

code for determining, for each [[said]] group, a portion of [[said]] the directed ~~adjacency~~ acyclic graph in accordance with activities of the operators, wherein the [[said]] portion of the directed ~~adjacency~~ acyclic graph is that portion which passes data up the directed ~~adjacency~~ acyclic graph; and

code for generating, for each [[said]] group, instructions for the determined portion of the directed ~~adjacency~~ acyclic graph, wherein operator instructions

are generated for those operators of the determined portion of the directed ~~adjacency~~ acyclic graph having active branches and wherein leaf instructions are generated for those graphic objects which are active at [[said]] the group of one or more pixel locations.

Claim 24 (currently amended): A computer readable medium ~~as claimed in~~ according to claim 23, wherein said computer program further comprises code for generating a table for storing data concerning the activity of the operators and the activity of the branches of the parent nodes.

Claim 25 (currently amended): A computer readable medium ~~as claimed in~~ according to claim 24, wherein said code for determining [[code]] groups comprises code for updating, for each [[said]] group, the [[said]] data stored in [[said]] the generated table.

Claim 26 (currently amended): A computer readable medium ~~as claimed in~~ according to claim 23, wherein said code for generating instructions comprises code for traversing [[said]] the portion of the directed ~~adjacency~~ acyclic graph and code for generating instructions for active operators and leaf value instructions in [[said]] the portion of the directed ~~adjacency~~ acyclic graph.

Claim 27 (currently amended): A computer readable medium ~~as claimed in~~ according to claim 23, wherein [[said]] the directed ~~adjacency~~ acyclic graph is an expression tree.

Claim 28 (currently amended): A computer readable medium ~~as claimed in~~ according to claim 23, wherein [[said]] the parent nodes represent binary operators.

Claim 29 (currently amended): A computer readable medium comprising a computer program for generating instructions for an expression tree, [[said]] the expression tree having a plurality of nodes comprising one or more binary nodes and a plurality of leaf nodes, wherein each [[said]] binary node ~~having~~ has a left branch to a descendent [[said]] node and a right branch to [[a]] another descendent [[said]] node and ~~representing~~ represents a binary operation on [[said]] the two descendant nodes, and wherein each [[said]] node represents a graphic object having object edges, with one or more [[said]] graphic objects overlapping, ~~each said~~ the overlapping graphics objects comprising a left node region, a common region, and a right node region, said computer program comprising:

code for determining groups of one or more pixel locations, wherein the groups are bounded by the object edges;

code for determining, for each [[said]] group, whether the left and right branches of [[said]] the one or more binary nodes are active or inactive;

code for traversing, for each [[said] group, [[said]] the expression tree, wherein the left branch of any previously traversed [[said]] binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] left node region is required for the binary operation of [[said]] the previously traversed binary node and the left branch is active and the right branch is inactive of [[said]] the previously traversed binary node, and wherein a right branch of any previously traversed binary node is ~~traversed to its said~~

~~descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] right node region is required for the binary operation of [[said]] the previously traversed binary node and the right branch is active and the left branch is inactive of [[said]] the previously traversed binary node; and

code for generating, for each [[said]] group, operator instructions for any [[said]] traversed binary node having active [[said]] right and left branches, and leaf value instructions for any traversed leaf node.

Claim 30 (currently amended): A computer readable medium ~~as claimed in~~ according to claim 29, wherein said ~~tranversing~~ traversing code traverses [[said]] the expression tree, wherein the left branch of any previously traversed [[said]] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node, and wherein the right branch of any previously traversed [[said]] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the right branch

of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node.

Claim 31 (currently amended): A computer readable medium comprising a computer program for rendering an expression tree into a raster pixel image having a plurality of scanlines and a plurality of pixel locations on each [[said]] scanline, [[said]] the expression tree having a plurality of nodes comprising one or more binary nodes and a plurality of leaf nodes, wherein each [[said]] binary node ~~having~~ has a left branch to a descendent [[said]] node and a right branch to [[a]] another descendent [[said]] node and ~~representing~~ represents a binary operation on [[said]] the two descendant nodes, and wherein each [[said]] node represents a graphic object having object edges, with one or more [[said]] graphic objects overlapping, ~~each said~~ the overlapping graphics objects comprising a left node region, a common region, and a right node region, said computer program comprising:

code for generating a table representative of [[said]] the expression tree, wherein [[said]] the table comprises a plurality of records corresponding to associated [[said]] binary nodes and leaf nodes, and each [[said]] record of ~~a said~~ an associated binary node comprises a first field indicating whether a [[said]] left region is required for operation of the operator of [[said]] the associated binary node, a second field indicating whether a right region is required for operation of the operator of [[said]] the associated binary node, a third field capable of indicating whether a [[said]] left branch of [[said]] the associated binary node is active, and a fourth field capable of indicating whether a [[said]] right branch of [[said]] the associated binary node is active;

code for determining groups of one or more pixel locations, wherein the groups are bounded by the object edges;

code for determining, for each [[said]] group, whether the left and right branches of [[said]] the one or more binary nodes are active or inactive;

code for updating, for each [[said]] group, [[said]] the third and fourth fields of [[said]] the table for [[said]] the determined active and inactive branches;

code for traversing, for each [[said]] group, [[said]] the expression tree in accordance with [[said]] the updated table wherein the left branch of any previously traversed [[said]] binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] left node region is required for the binary operation of [[said]] the previously traversed binary node and the left branch is active and the right branch is inactive of [[said]] the previously traversed binary node, and wherein a right branch of any previously traversed binary node is ~~traversed to its said descendent node if~~ ignored unless the right and left branches of [[said]] the previously traversed binary node are active or [[if]] a [[said]] right node region is required for the binary operation of [[said]] the previously traversed binary node and the right branch is active and the left branch is inactive of [[said]] the previously traversed binary node;

code for generating, for each [[said]] group, operator instructions for any [[said]] traversed binary node having active [[said]] right and left branches, and leaf value instructions for any traversed leaf node; and

code for executing, for each [[said]] group, corresponding [[said]] generated instructions so as to render [[said]] the image.

Claim 32 (currently amended): A computer readable medium ~~as claimed in~~ according to claim 31, wherein [[said]] the table further comprises for each [[said]] record of ~~a said~~ an associated binary node a fifth field indicating whether the graphic object representing [[said]] the descendent node on the right branch of [[said]] the associated binary node obscures the graphic object representing [[said]] the descendent node on the left branch of [[said]] the associated binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the associated binary node, and a sixth field indicating whether the graphic object representing [[said]] the descendent node on the left branch of [[said]] the associated binary node obscures the graphic object representing [[said]] the descendent node on the right branch of [[said]] the associated binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the associated binary node[[;]], and

wherein said ~~tranversing~~ traversing means traverses [[said]] the expression tree in accordance with [[said]] the updated table, wherein the left branch of any previously traversed [[said]] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node, and wherein the right branch of any previously traversed [[said]] binary node is traversed to its [[said]] descendent node if the right and left branches of [[said]] the previously traversed binary node are active and if the

graphic object representing [[said]] the descendent node on the left branch of [[said]] the previously traversed binary node does not obscure the graphic object representing [[said]] the descendent node on the right branch of [[said]] the previously traversed binary node in the common region of the graphic objects representing [[said]] the descendent nodes of [[said]] the previously traversed binary node.

Claim 33 (currently amended): A computer program ~~as claimed in~~ according to claim 32, wherein [[said]] the fifth and sixth fields are used to implement a CLIP IN or a CLIP OUT operation.